AMENDMENTS TO THE CLAIMS

1. (Original) A computer-implemented process for receiving media data across a firewall, comprising the process actions of:

receiving an Internet client's encrypted media packet sent using Real-time Transport Protocol (RTP) message format at a media-relay server;

retrieving a sending client's Security Association (SA) using the source information included in the RTP message header,

if no SA exists, dropping the media packet at the media-relay server;

if a SA does exist, making a copy of the encrypted media packet and decrypting the media packet;

obtaining a Synchronization Source Identifier (SSRC) from the SA;

using the Synchronization Source Identifier included in the decrypted RTP packet and comparing it with the Synchronization Source Identifier obtained from the SA;

if the Synchronization Source Identifier included in the decrypted RTP packet does not match the Synchronization Source Identifier obtained from the SA, dropping the media packet; and

if the Synchronization Source Identifier in the decrypted RTP packet matches to the Synchronization Source Identifier obtained from the SA, forwarding the packet to a network client.

2. (Original) The computer-implemented process of Claim 1 wherein the source information retrieved by the media-relay server comprises a source Internet Protocol (IP) address and port number found in the RTP message format.

3. (Original) The computer-implemented process of Claim 1 wherein the media packet comprises audio data.

4. (Original) The computer-implemented process of Claim 1 wherein the media packet comprises video data.

5. (Original) A computer-implemented process for receiving media data across a firewall, comprising the process actions of:

receiving a sending client's encrypted media packet at a first media-relay server;

said first media-relay server forwarding said media packet to a second media-relay server;

said second media-relay server, retrieving a sending client's Security Association (SA) using a Synchronization Source Identifier appended to the media packet that is not encrypted;

if no such SA exists, dropping the media packet;

if such a SA does exist, making a copy of the media packet;

decrypting the packet;

comparing the Synchronization Source Identifier inside the decrypted media packet with the Synchronization Source Identifier appended to the media packet,

if the Synchronization Source Identifier inside the decrypted media packet does not match the Synchronization Source Identifier appended to the media packet, dropping the media packet;

if the Synchronization Source Identifier inside the decrypted media packet matches the Synchronization Source Identifier appended to the media packet, forwarding the packet is forwarded to a corporate client.

6. (Original) The computer-implemented process of Claim 5 wherein the sending client sends the media packet via RTP using an RTP header, and wherein the first media-relay server modifies the RTP header to include the appended Synchronization Source Identifier concatenated with the RTP header prior to forwarding the media packet to the second media-relay server.

7. (Original) The computer-implemented process of Claim 6 wherein the media packet is transferred by opening only two User Datagram Protocol (UDP) ports on an external firewall and multiple UDP ports on an internal firewall.

8. (Original) The computer-implemented process of Claim 5 wherein the sending client sends the media packet to the first media-relay server after modifying the RTP header to include an appended Synchronization Source Identifier concatenated with the RTP header.

9. (Original) The computer-implemented process of Claim 8 wherein the first media-relay server sends the modified RTP header with the appended Synchronization Source Identifier to the second media relay server.

10. (Original) The computer-implemented process of Claim 9 wherein the media packet is transferred by opening two UDP ports on an external firewall and two UDP ports of an internal firewall.

11. (Original) The computer-implemented process of Claim 5 wherein the first media relay server is in a Demilitarized Zone of a network and a third media-relay server is in the internal network, and wherein the media packet is sent from the first media relay server to the third media-relay server before sending the media packet to the second media-relay server in a different network from the first media-relay server and the third media-relay server.

12. (Original) The computer-implemented process of Claim 11 wherein the first media relay server and the third media relay server communicate using Transmission Control Protocol (TCP).

13. (Original) The computer-implemented process of Claim 12 wherein the media packet is transferred by opening two UDP ports on an external firewall and one TCP port on an internal firewall.

14. (Original) The computer-implemented process of Claim 5 wherein the first media server assigns the Synchronization Source Identifier to the sending client.

15. (Original) A data structure for access by an application program being executed on a data processing system, comprising:

an unencrypted Synchronization Source Identifier concatenated with an encrypted RTP header containing a Synchronization Source Identifier; and

an encrypted media data packet.

16. (Original) A system for formatting data to traverse at least one firewall, comprising:

a first media-relay server assigning a Synchronization Source Identifier to a sending client;

receiving a sending client's encrypted media packet via RTP at the first media-relay server;

said first media-relay server forwarding said encrypted media packet to a second media-relay server with said assigned Synchronization Source Identifier appended to the encrypted media packet;

said second media-relay server, retrieving the sending client's Security Association (SA) using a Synchronization Source Identifier appended to the encrypted media packet;

if no such SA exists, dropping the media packet;

if such a SA does exist, making a copy of the media packet;

decrypting the packet;

comparing the Synchronization Source Identifier inside the decrypted media packet with the Synchronization Source Identifier appended to the media packet, and

if the Synchronization Source Identifier inside the decrypted media packet does not match the Synchronization Source Identifier appended to the media packet, dropping the media packet;

if the Synchronization Source Identifier inside the decrypted media packet matches the Synchronization Source Identifier appended to the media packet, forwarding the media packet to a network client.